REMARKS

Reconsideration and allowance are respectfully requested.

The IDS filed on November 19, 2009 has not been formally acknowledged by the Examiner. Applicants respectfully request consideration and acknowledgement.

The independent claims are amended. Example support for (1) the random secret being "unknown after being stored in the storage device or option" may be found at page 15, lines 19-20 and page 19, lines 17-22 of the specification and (2) for generating a temporarily available instance of the device-specific security data internally confined within said electronic circuit during usage of said device and "not stored in a memory such that the temporarily available instance of the device-specific security data is only available as long as the externally received trigger data is received" may be found at page 22, lines 8-13, page 24, lines 1-3, and page 24, lines 1-3 and 13-15 of the specification.

Claims 47, 48, 52, 54, 58-61, 66, 67, 69-71, 73, and 75 stand rejected for obviousness based on newly-applied Kocher (US20020099948), newly-applied Richards (US20010054147), and newly-applied Knapen (US20030053629). This rejection is respectfully traversed.

The Examiner relies primarily on Kocher's paragraphs 78, 81, and 104 along with claim 1 of Kocher. The Examiner reads Kocher's "group verification result" (presumably the key distribution message KDM associated with the content as described in [0068-69] onto the claimed "trigger data," but it is unclear what in Kocher the Examiner maps to the claimed random secret. Perhaps the Examiner is trying to map Kocher's CHIP_KEY and BATCH_KEY to the claimed random secret?

Kocher discloses a cryptographic rights unit CRU 225 which performs cryptographic operations in order to access digital content. During a "personalization" process, a device-

specific key CHIP_KEY and a group key BATCH_KEY are loaded into memory 265 of the CRU. As explained in [0081], personalization is performed during manufacturing of the device. Kocher then adds rights to specific content by sending rights enablement messages REMs that include encrypted rights key "fixup values." The encrypted rights key values are sent to the CRU and are decrypted using CHIP_KEY as described in [0076] and [0080]. The processed (by a function F) encrypted rights key is stored in memory at address selected by interface control processor ICP 235 which controls reading and writing into the protected memory.

Thus, after personalization, security data is stored in the CRU related to specific contents. In contrast, the claimed device-specific security data in claim 47 is not stored in a protected memory 265 like it is in Kocher. Instead, the claimed device-specific security data is only temporarily available as an instance of the device-specific security data when externally-received trigger data, previously generated during configuration of the tamper-resistant electronic circuit using the random secret and device-specific security data that is different from the random secret, is externally received by the cryptographic processing engine.

The Examiner appears to map Kocher's key distribution message KDM to the claimed trigger data. KDM includes a content description key CDK generator (an encrypted CDK). At an appropriate address to protected memory, a rights key is retrieved for processing (decryption) of the CDK generator for generation of the CDK. The trigger data likewise is a representation (basically encryption) of content key for decryption of content. Kocher is concerned with distribution of content. The process may be complex if distribution is made to a large group of content receivers. To reduce complexity, Kocher incorporates a group key called a BATCH_KEY (see e.g. [0075]) and verifies that a CRU is included in an authorized group of entities (see e.g. [0080]). The Examiner incorrectly interprets the verification result as the

claimed trigger data. Claim 1 in Kocher, for example, recites at step (c) that the verification result only verifies that an addressed device belongs to a specified group. In response to successful verification, a group key is retrieved from memory. Thus, the verification result as such cannot be reasonably interpreted as the claimed trigger data.

Kocher is concerned with distributing encrypted contents to users with play-back devices. Devices for specific content has been paid for are configured with security information (a rights key) related to the purchased content. The management of the rights keys in the communication between content provider and user device critically depends on the CHIP_KEY which is known to the content provider. But as recited in claim 47, the random secret (to which Kocher's CHIP_KEY is mapped) is not known to the content provider: "a random secret not accessible over any external circuit interface to the tamper-resistant electronic circuit <u>unknown after being stored in the storage device or option</u>." If anything, CHIP_KEY more closely corresponds to device-specific data. But unlike the stored CHIP_KEY, the claimed device-specific data is not stored but instead is reproduced on-the-fly when the trigger data is received.

Although the claimed technology may be used to decrypt received encrypted content, it is not developed for a content distribution scenario like Kocher's system. A content provider knowing the device-specific data may encrypt content using this known device-specific data and provide user with corresponding information for regeneration in device-specific data circuit to decrypt the content. The same key (i.e., the device-specific data) is re-used any time that content is distributed in Kocher.

So another missing feature is the claimed random secret. Kocher's personalizing function must know CHIP_KEY in order to generate encrypted rights key. As explained above, the claimed random secret is "unknown after being stored in the storage device or option."

There is a security risk of generating and/or storing cryptographic keys on an IC on behalf of a device manufacturer or content provider. The claimed technology eliminates the need to communicate the random secret from the IC manufacturer to a device manufacturer by allowing an IC to be configured without knowing the random secret stored in the IC. During configuration of the IC, an administrator simply inputs selected device specific data and records generated trigger output.

Another advantage is that even though a break into the IC in Kocher's system may reveal secrets such as the rights keys and device-specific key CHIP_KEY, a break into an IC using the claimed technology does not reveal the claimed random secret. If the random secret were somehow revealed, the device-specific data can not be determined without having the trigger data.

The Examiner admits that Kocher lacks outputting the trigger data and a teaching that the security data is temporary. The Examiner points to Knapen for temporary keys based on received triggering data citing the Abstract and paragraphs 17, 21-23. In [0017], Knapen teaches a new-key scheduler 110 that triggers generation of a new key by key selector 150. But Applicants do not see in the Abstract and paragraphs 17, 21-23 a teaching by Knapen that the new key is "only available as long as the externally received trigger data is received."

Knapen distributes contents in pieces where each content piece is separately encrypted. Information associated with these pieces is sent to the receiver which may generate corresponding decryption keys for decrypting the separately-encrypted content pieces. In particular, section [0017], referred to by the Examiner describes a new-key scheduler 110 that triggers generation of a new key by key selector 150 for encryption of content pieces. The term "trigger" is used by Knapen as a signal that initiates generation of a key. But the trigger signal is

not part of the generating process but merely acts as an initiating entity. In contrast, the claimed trigger data is actively involved in the generation of a temporarily-available instance of device specific security data internally confined within the circuit.

Richards discloses method for authentication based on a public key. Paragraph [0035] describes details of the authentication method: a host generates a random number, stores it, encrypts the random number using a user's public key, and sends encrypted number to user. User decrypts the received encrypted number using the private key and returns the random number to host which compares the received random number with the stored random number. Authentication is verified if the compared numbers match. There is no teaching of the claimed trigger data. The claims implement and utilize device-specific security data which is different from authentication.

For the reasons explained above, the obviousness rejection should be withdrawn. If the Examiner elects to make any further prior art rejection, the Examiner is requested to specifically identify in the prior art references what information in each reference in each paragraph corresponds to the claimed random secret, trigger data, device-specific security data that is different from the random secret, the manufacturing/configuring operations, the use operation, and the temporarily available instance of the device-specific security data. Although the Examiner identifies paragraphs, it is uncertain what specifically in those paragraphs the Examiner maps to each element, e.g., "trigger data generating circuitry," and each claim term, e.g., "random secret."

The application is in condition for allowance. An early notice to that effect is requested.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: ____________________
John R. Lastova
Reg. No. 33,149

JRL:maa
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100